C 10

```
EEEEEEEEEEEEEEE  DDDDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD      TTTTTTTTTTTTTTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEEEEEEEEEEE     DDD         DDD         TTT
EEEEEEEEEEEE     DDD         DDD         TTT
EEEEEEEEEEEE     DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEE              DDD         DDD         TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD            TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD            TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD            TTT
```

```
**FILE**ID**CHMPAREN                                   C  5



  CCCCCCCC  HH      HH  MM      MM  PPPPPPPP      AAAAAA    RRRRRRRR    EEEEEEEEEE  NN      NN
  CCCCCCCC  HH      HH  MM      MM  PPPPPPPP      AAAAAA    RRRRRRRR    EEEEEEEEEE  NN      NN
CC          HH      HH  MMMM  MMMM  PP      PP  AA      AA  RR      RR  EE          NN      NN
CC          HH      HH  MMMM  MMMM  PP      PP  AA      AA  RR      RR  EE          NN      NN
CC          HH      HH  MM  MM  MM  PP      PP  AA      AA  RR      RR  EE          NNNN    NN
CC          HH      HH  MM  MM  MM  PP      PP  AA      AA  RR      RR  EE          NNNN    NN
CC          HHHHHHHHHH  MM      MM  PPPPPPPP    AA      AA  RRRRRRRR    EEEEEEEE    NN  NN  NN
CC          HHHHHHHHHH  MM      MM  PPPPPPPP    AA      AA  RRRRRRRR    EEEEEEEE    NN  NN  NN
CC          HH      HH  MM      MM  PP          AAAAAAAAAA  RR  RR      EE          NN    NNNN
CC          HH      HH  MM      MM  PP          AAAAAAAAAA  RR  RR      EE          NN    NNNN
CC          HH      HH  MM      MM  PP          AA      AA  RR    RR    EE          NN      NN    ....
CC          HH      HH  MM      MM  PP          AA      AA  RR    RR    EE          NN      NN    ....
  CCCCCCCC  HH      HH  MM      MM  PP          AA      AA  RR      RR  EEEEEEEEEE  NN      NN    ....
  CCCCCCCC  HH      HH  MM      MM  PP          AA      AA  RR      RR  EEEEEEEEEE  NN      NN    ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

EDT
V04

```
0 %TITLE 'EDT$CHMPAREN - handle parentheses'
0 MODULE EDT$CHMPAREN (                                    ! Handle parentheses
0               IDENT = 'V04-000'                                    ! File: CHMPAREN.BLI Edit: JBS1007
0               ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:     EDT -- The DEC Standard Editor
1 !
1 ! ABSTRACT:
1 !
1 !       This module implements a parenthesized string of commands.
1 !
1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
1 !
1 ! AUTHOR: Bob Kushlis, CREATION DATE: Unknown
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  DJS 04-Feb-1981.  This module was created by
1 !       extracting the routine EDT$$PARENT  from module CHANGE.BLI.
1 ! 1-002 - Regularize headers.  JBS 03-Mar-1981
1 ! 1-003 - Use new message codes.  JBS 04-Aug-1981
1 ! 1-004 - Add return values.  JBS 02-Oct-1981
1 ! 1-005 - Add an entry point so that this module can be loaded back into
1 !         memory after calling the change mode parser.  Also, use a different
1 !         symbol for the name of the change mode parser so the call can be
1 !         ignored by the overlay analyzer (and so that the call can be
1 !         intercepted by a routine which will load us back into memory).
1 !         JBS 02-Apr-1982
1 ! 1-006 - Set a flag if control C actually aborts something.  JBS 24-May-1982
1 ! 1-007 - If an error occurs in processing the commands inside parens, such
1 !         as string not found, continue after the right paren.  JBS 23-Feb-1983
```

EDT
V04

```
;  58          0058  1 !--
;  59          0059  1
;    61          0060   1 %SBTTL 'Declarations'
;    62          0061   1 !
;    63          0062   1 ! TABLE OF CONTENTS:
;    64          0063   1 !
;    65          0064   1
;    66          0065   1 REQUIRE 'EDTSRC:TRAROUNAM';
;    67          0504   1
;    68          0505   1 FORWARD ROUTINE
;    69          0506   1     EDT$$PARENT,                                    ! Collect and execute a parenthesized string of commands
;    70          0507   1     EDT$$LOAD_CHMPAREN : NOVALUE;                   ! Load this module into memory (on the PDP-11)
;    71          0508   1
;    72          0509   1 !
;    73          0510   1 ! INCLUDE FILES:
;    74          0511   1 !
;    75          0512   1
;    76          0513   1 REQUIRE 'EDTSRC:EDTREQ';
;    77          0648   1
;    78          0649   1 !
;    79          0650   1 ! MACROS:
;    80          0651   1 !
;    81          0652   1 !        NONE
;    82          0653   1 !
;    83          0654   1 ! EQUATED SYMBOLS:
;    84          0655   1 !
;    85          0656   1 !        NONE
;    86          0657   1 !
;    87          0658   1 ! OWN STORAGE:
;    88          0659   1 !
;    89          0660   1 !        NONE
;    90          0661   1 !
;    91          0662   1 ! EXTERNAL REFERENCES:
;    92          0663   1 !
;    93          0664   1 !        In the routine
;  1 %SBTTL 'EDT$$PARENT  - handle parentheses'
;  1
;  1 GLOBAL ROUTINE EDT$$PARENT (                           ! Handle parentheses
;  1     COUNT,                                             ! Number of times to execute
;  1     EXECUTE                                            ! 1 = execute the commands
;  1     ) =
;  1
;  1 !++
;  1 ! FUNCTIONAL DESCRIPTION:
;  1 !
;  1 !       This routine implements a parenthesized string of commands.  The
;  1 !       string is analyzed without execution in order to verify it syntactically,
;  1 !       and find the end of the string.  It is then executed COUNT times, or until
;  1 !       a command in the string fails.  In either case, EDT$$A_CMD_BUF is left pointing
;  1 !       at the first character after the parenthesized command list.
;  1 !
;  1 ! FORMAL PARAMETERS:
;  1 !
;  1 !  COUNT                Number of times to execute
;  1 !
;  1 !  EXECUTE              1 = execute the commands
;  1 !
;  1 ! IMPLICIT INPUTS:
;  1 !
;  1 !       EDT$$A_CMD_BUF
;  1 !
;  1 ! IMPLICIT OUTPUTS:
;  1 !
;  1 !       EDT$$G_CC_DONE
;  1 !
;  1 ! ROUTINE VALUE:
;  1 !
;  1 !       1 = ok, 0 = control C, 2 = end of journal file
;  1 !
;  1 ! SIDE EFFECTS:
;  1 !
;  1 !       NONE
;  1 !
;  1 !--
;  1
;  2     BEGIN
;  2
;  2     EXTERNAL ROUTINE
;  2         EDT$$MSG_BELL : NOVALUE,                       ! Output a message to the terminal with a warning bell
;  2         EDT$$CHK_CC,                                   ! Check to see if a CTRL/C has been typed
;  2         EDT$$CHM_PAREXE_NOOVERLAY;                     ! Parse and execute a change mode command string
;  2
;  2     EXTERNAL
;  2         EDT$$A_CMD_BUF,                                ! Command string pointer
;  2         EDT$$G_CC_DONE;                                ! Set to 1 if control C actually aborts something
;  2
;  2     MESSAGES ((PARENMIS));
;  2
;  2     LOCAL
;  2         SUCCEED,
;  2         COM_END,
;  2         COM_START;
```

```
2
2      EDT$$A_CMD_BUF = COM_START = CH$PLUS (.EDT$$A_CMD_BUF, 1);
2  !+
2  ! Check syntax and find the end of the command.
2  !-
2      SUCCEED = EDT$$CHM_PAREXE_NOOVERLAY (0);
2
2      IF (.SUCCEED NEQ 1) THEN RETURN (.SUCCEED);
2
2      COM_END = .EDT$$A_CMD_BUF;
2  !+
2  ! If we are not positioned on the right paren we have a syntax error.
2  !-
2
3      IF (CH$RCHAR (.COM_END) NEQ %C')')
2      THEN
3          BEGIN
3          EDT$$MSG_BELL (EDT$_PARENMIS);
3          RETURN (0);
2          END;
2
2      IF .EXECUTE
2      THEN
2
2          DECR I FROM .COUNT - 1 TO 0 DO
3              BEGIN
3
3              IF EDT$$CHK_CC ()
3              THEN
4                  BEGIN
4                  EDT$$G_CC_DONE = 1;
4                  RETURN (0);
3                  END;
3
3              EDT$$A_CMD_BUF = .COM_START;
3              SUCCEED = EDT$$CHM_PAREXE_NOOVERLAY (1);
3
3              IF (.SUCCEED NEQ 1) THEN EXITLOOP;
3
2              END;
2
2      EDT$$A_CMD_BUF = CH$PLUS (.COM_END, 1);
2  !+
2  ! Don't indicate an error in executing the command within parens as an error
2  ! in executing this command.  This allows multiple commands to be executed
2  ! terminated by search failures.
2  !-
2
2      IF (.SUCCEED EQL 0) THEN SUCCEED = 1;
2
2      RETURN (.SUCCEED);
1      END;                                        ! of routine EDT$PARENT

                                                    .TITLE  EDT$CHMPAREN EDT$CHMPAREN - handle parentheses
                                                    .IDENT  \V04-000\
                                                             .EXTRN  EDT$$MSG_BELL, EDT$$CHK_CC
                                                             .EXTRN  EDT$$CHM_PAREXE_NOOVERLAY
                                                             .EXTRN  EDT$$A_CMD_BUF, EDT$$G_CC_DONE
                                                             .EXTRN  EDT$_PARENMIS

                                                             .PSECT  _EDT$CODE,NOWRT, SHR, PIC,2

                                    00FC 00000               .ENTRY  EDT$$PARENT, Save R2,R3,R4,R5,R6,R7              ; 0667
                57 00000000G  00      9E 00002               MOVAB   EDT$$CHM_PAREXE_NOOVERLAY, R7                    ;
                56 00000000G  00      9E 00009               MOVAB   EDT$$A_CMD_BUF, R6                               ;
      54        66            01      C1 00010               ADDL3   #1, EDT$$A_CMD_BUF, COM_START                    ; 0723
                66            54      D0 00014               MOVL    COM_START, EDT$$A_CMD_BUF                        ;
                              7E      D4 00017               CLRL    -(SP)                                            ; 0727
                67            01      FB 00019               CALLS   #1, EDT$$CHM_PAREXE_NOOVERLAY                    ;
                53            50      D0 0001C               MOVL    R0, SUCCEED                                      ;
                01            53      D1 0001F               CMPL    SUCCEED, #1                                      ; 0729
                              52      12 00022               BNEQ    6$                                               ;
                55            66      D0 00024               MOVL    EDT$$A_CMD_BUF, COM_END                          ; 0731
                29            65      91 00027               CMPB    (COM_END), #41                                   ; 0736
                              0F      13 0002A               BEQL    1$                                               ;
                   00000000G  8F      DD 0002C               PUSHL   #EDT$_PARENMIS                                   ; 0739
      00000000G 00            01      FB 00032               CALLS   #1, EDT$$MSG_BELL                                ;
                              3F      11 00039               BRB     7$                                               ; 0740
                2C        08  AC      E9 0003B  1$:          BLBC    EXECUTE, 5$                                      ; 0743
                52        04  AC      D0 0003F               MOVL    COUNT, I                                         ; 0756
                              23      11 00043               BRB     4$                                               ;
      00000000G 00            00      FB 00045  2$:          CALLS   #0, EDT$$CHK_CC                                  ; 0749
                09            50      E9 0004C               BLBC    R0, 3$                                           ;
      00000000G 00            01      D0 0004F               MOVL    #1, EDT$$G_CC_DONE                               ; 0752
                              22      11 00056               BRB     7$                                               ; 0753
                66            54      D0 00058  3$:          MOVL    COM_START, EDT$$A_CMD_BUF                        ; 0756
                              01      DD 0005B               PUSHL   #1                                               ; 0757
                67            01      FB 0005D               CALLS   #1, EDT$$CHM_PAREXE_NOOVERLAY                    ;
                53            50      D0 00060               MOVL    R0, SUCCEED                                      ;
                01            53      D1 00063               CMPL    SUCCEED, #1                                      ; 0759
                              03      12 00066               BNEQ    5$                                               ;
                DA            52      F4 00068  4$:          SOBGEQ  I, 2$                                            ; 0746
                66        01  A5      9E 0006B  5$:          MOVAB   1(R5), EDT$$A_CMD_BUF                            ; 0763
                              53      D5 0006F               TSTL    SUCCEED                                          ; 0770
                              03      12 00071               BNEQ    6$                                               ;
                53            01      D0 00073               MOVL    #1, SUCCEED                                      ;
                50            53      D0 00076  6$:          MOVL    SUCCEED, R0                                      ; 0772
                                      04 00079               RET                                                      ;
                              50      D4 0007A  7$:          CLRL    R0                                               ; 0773
                                      04 0007C               RET                                                      ;

; Routine Size:  125 bytes,    Routine Base:  _EDT$CODE + 0000


;  204          0774  1
;  206          0775  1 %SBTTL 'EDT$$LOAD_CHMPAREN - load this module into memory'
;  207          0776  1
;  208          0777  1 GLOBAL ROUTINE EDT$$LOAD_CHMPAREN                   ! Load this module into memory
;  209          0778  1     : NOVALUE =
;  210          0779  1
;  211          0780  1 !++
;  212          0781  1 ! FUNCTIONAL DESCRIPTION:
;  213          0782  1 !
;  214          0783  1 !       This routine has no function.  It exists as an entry point so that
;  215          0784  1 !       EDT$$CHM_PAREXE_NOOVERLAY can call this module back into memory before
;  216          0785  1 !       returning to it.
;  217          0786  1 !
;  218          0787  1 ! FORMAL PARAMETERS:
;  219          0788  1 !
;  220          0789  1 !       NONE
;  221          0790  1 !
;  222          0791  1 ! IMPLICIT INPUTS:
;  223          0792  1 !
;  224          0793  1 !       NONE
;  225          0794  1 !
;  226          0795  1 ! IMPLICIT OUTPUTS:
;  227          0796  1 !
;  228          0797  1 !       NONE
;  229          0798  1 !
;  230          0799  1 ! ROUTINE VALUE:
;  231          0800  1 !
;  232          0801  1 !       NONE
;  233          0802  1 !
;  234          0803  1 ! SIDE EFFECTS:
;  235          0804  1 !
;  236          0805  1 !       NONE
;  237          0806  1 !
;  238          0807  1 !--
;  239          0808  1
;  240          0809  2    BEGIN
;  241          0810  2    0
;  242          0811  1    END;                                         ! of routine EDT$$LOAD_CHMPAREN



                                       0000 00000        .ENTRY  EDT$$LOAD_CHMPAREN, Save nothing              ; 0777
                                         04 00002        RET                                                   ; 0811

; Routine Size:  3 bytes,    Routine Base:  _EDT$CODE + 007D


;  243          0812  1
;  244          0813  1 !<BLF/PAGE>
;  246          0814  1 END                                                 ! of module EDT$CHMPAREN
;  247          0815  1
;  248          0816  0 ELUDOM



;                                PSECT SUMMARY
;
;       Name                      Bytes                       Attributes
;
;  _EDT$CODE                           128  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)



;                         Library Statistics
;
;                                             -------- Symbols --------       Pages       Processing
;       File                                  Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[EDT.SRC]EDT.L32;1               377         1         0          40        00:00.2
;  _$255$DUA28:[EDT.SRC]PSECTS.L32;1              2         1        50           7        00:00.1



;                                 COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:CHMPAREN/OBJ=OBJ$:CHMPAREN MSRC$:CHMPAREN.BLI/UPDATE=(ENH$:CHMP
;       AREN)

; Size:           128 code + 0 data bytes
; Run Time:          00:12.7
; Elapsed Time:      00:16.3
; Lines/CPU Min:       3855
; Lexemes/CPU-Min: 10502
; Memory Used:   79 pages
; Compilation Complete
```

0132 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY